Claims: We claim:

1) A method of increasing the router fanout in a redundant multi-stage network which minimizes the impact on throughput bandwidth,

where the multi-stage network has a plurality of rows,

where the rows have a plurality of routers,

where each router has a plurality of ports,

where the number of new routers is equal to the number of rows, and

where external port is any top port on a new router in the first row or any bottom port on a new router in the last row

comprising the step of:

a) selecting a previously unselected row in which to insert a new router;
b) selecting the position within the selected row to insert the new router;
c) inserting the new router in the selected position within the selected row;
d) repeating steps a), b), and c) for all the rows which have not been previously selected;
e) rewiring any port of any router in any row which is not connected to the proper port of the proper router;
f) repeating step e) for all the ports which are not connected to the proper port of the proper router and have not been previously selected;
g) connecting all external ports of new routers in the first row and the last row; and
h) activating all external ports of new routers in the first row and the last row.

2) The method as claimed in Claim 1, wherein the step e) is replaced by a new step e)

e) either rewiring any port of any router in any row which is not connected to the proper port of the proper router or permuting the labeling of the router ports of any router to reduce the number of connections which have to be rewired;

whereby reducing the number of connections that have to be rewired and thus reducing the reducing the effect on the throughput bandwidth.

3) A method of increasing the width of a redundant multi-stage network which minimizes the impact on throughput bandwidth,

where the multi-stage network has a plurality of rows,

where the rows have a plurality of routers,

where each router has a plurality of ports,

where the number of new routers is equal to the number of rows, and

where external port is any top port on a new router in the first row or any bottom port on a new router in the last row

comprising the steps of:

a) selecting a previously unselected row in which to insert a new router;
b) selecting the position within the selected row to insert the new router;
c) inserting the new router in the selected position within the selected row;
d) rewiring the connections to and from the selected row which are not connected to the proper port of the proper router in the selected row;
e) repeating the previous steps for all the rows which have not been previously selected;
f) connecting all external ports of new routers in the first row and the last row; and
g) activating all external ports of new routers in the first row and the last row.

4) The method as claimed in Claim 3, wherein the step

h) permuting the labeling of the router ports to reduce the number of connections which have to be rewired;

is inserted between steps c) and d), whereby reducing the number of connections that have to be rewired and thus reducing the reducing the effect on the throughput bandwidth.

5) The selection of a previously unselected row as claimed in step a) of Claim 3, wherein the row is the middle row or closest to the middle; whereby the selected row has the greatest topological redundancy, thus reducing the effect of any rewiring on the throughput bandwidth.

6) The method of Claim 3, wherein

all the new routers are first connected to form a column with the same number of rows as the multi-stage network;

the new router in step b) is a column router with a row which corresponds to the selected row in step a); and

the position selected in step b) is at the same end of the row as the previously selected new routers;

whereby the first connected column routers reduce the number of connection that have to be rewired and thus reduce the impact on the throughput bandwidth.

7) The rewiring of the connections to and from the selected row which are not connected to the proper port of the proper node in the selected row as claimed in step d) of Claim 3, wherein ports not currently connected are given priority; whereby reducing the number of disconnected connections at any given time and thus reducing the impact on throughput bandwidth.

8) A method of increasing the width of a redundant multi-stage network which minimizes the impact on throughput bandwidth as shown in Fig. 14,

where the multi-stage network has a plurality of rows,

where the rows have a plurality of routers,

where the routers have a plurality of ports,

where current_row is a variable which indicates a row of the multi-stage network,

where N is a variable which indicates a column of the multi-stage network,

where number_of_rows is a constant equal to the number of rows in the multi-stage network,

where R(row, column) is a variable which indicates a router in a multi-stage network, and

where insertion_position(x) is a function which indicates the position in a row a new router should be inserted

where external port is any top port on a new router in the first row or any bottom port on a new router in the last row

comprising the steps of:

a) begin,
b) set current_row to 0,
c) insert router R(current_row, N) into position insertion_position(current_row),
d) if current_row > 1 then go to step e else go to step g,
e) if bottom port of router R(current_row-1, N) is connected to any top port of router R(current_row, N) then go to step f else got to step g,
f) connect bottom port of router R(current_row-1, N) to top port router R(current_row, N),
g) increment current_row,
h) if current_row < number_of_rows) then go to step c else go to step i,
i) if there are any misconnected ports then go to step l else go to step j,
j) connect and activate all disconnected external ports,
k) end
l) if there are any ports that are not connected to their appropriate ports then go to step m else go to step j,

m) select port not connected to its appropriate port and call it corresponding_port,

n) if current_port is already connected then go to step o else go to step p,

o) disconnect current_port from existing connection,

p) if corresponding_port is already connected then go to step q else go to step r,

q) disconnect corresponding_port from existing connection,

r) connect current_port to corresponding_port and go to step i.

9) The method as claimed in steps a) through r) of Claim 8, wherein steps

t) if there are any routers whose ports can be relabeled then go to step u else go to step l,

u) relabel the appropriate ports in that router and go to step i,

are added and step i is replaced by a new step i,

i) if there are any misconnected ports then go to step t else go to step j,

whereby the router ports are examined to see if a connection can be avoided by re-labeling the ports.

10) A method of increasing the width of a redundant multi-stage network which minimizes the impact on throughput bandwidth as shown in Fig. 15,

where the multi-stage network has a plurality of rows,

where the rows have a plurality of routers,

where the routers have a plurality of ports,

where current_row is a variable which indicates a row of the multi-stage network,

where N is a variable which indicates a column of the multi-stage network,

where number_of_rows is a constant equal to the number of rows in the multi-stage network,

where R(row, column) is a variable which indicates a router in a multi-stage network,

where insertion_position(x) is a function which selects the position in a row a new router should be inserted,

where rindex is a variable which indicates a row of the multi-stage network,

where row_select(x) is a function which selects a row in the multi-stage network,

where current_port is a variable which indicates a router port,

where port_select(x) is a function which selects a router port, and

where corresponding_port is a variable which indicates a router port, and

where external port is any top port on a new router in the first row or any bottom port on a new router in the last row

comprising the steps of:

a) begin
b) set current_row to 0,
c) insert router R(current_row, N) into position insertion_position(current_row),
d) if current_row > 1 then go to step e else go to step g,
e) if bottom port of router R(current_row-1, N) is connected to top port of router R(current_row, N) then go to step f else go to step g,
f) connect bottom port of router R(current_row-1, N) to top port router R(current_row, N),
g) increment current_row,
h) if current_row < number_of_rows) then go to step c else go to step i,

i) set rindex to 0,

j) set current_row to row_select(rindex),

k) set current_port to port picked by port_select(x),

l) set corresponding_port to the port that current_port should be connected to,

m) if there are more ports to select then go to step n else go to step t,

n) if current_port already connected then go to step o else go to step p,

o) disconnect current_port from existing connection,

p) if current_port is already connected then go to step q else go to step r,

q) disconnect corresponding_port from existing connection,

r) connect current_port to corresponding_port,

s) increment rindex, and

t) if rindex < number_of_rows then got to step j else go to step u,

u) connect and activate all disconnected external ports, and

v) end.

11) The method as claimed in steps a) through v) of Claim 10, wherein step

w) relabel ports of current_row

is inserted between step j and step k, whereby the router ports are relabeled to avoid the need to make a connection.

12) The port relabeling method as claimed in step w of Claim 11 as shown in Fig. 18,

where bport is a bottom port of a router,

where tport is a top port of a router,

where dest_router is a router, and

where dest_port is a port,

wherein the port relabeling method comprises the steps:

a) begin
b) set bport to left most bottom router of the row
c) set source router to be the one bport belongs to
d) set dest_router to be the one bport is connected to
e) if bport is supposed to be connected to dest_router then go to step f else go to step h,
f) if any port on source router is supposed to be connected to dest_router then go to step g else go to step h,
g) exchange bport with that port,
h) if there is a bottom port to the right of bport then go to step i else go to step j,
i) set bport to that bottom port right of bport and go to step d,
j) set tport to right most bottom router of the row,
k) set source router to be the one tport belongs to,
l) set dest_port to be the one tport is connected to,
m) if tport is supposed to be connected dest_port then go to step n else go to step p,
n) if any port on source router is supposed to be connected to tport then go to step o else go to step p,
o) exchange tport with that port and go to step p,
p) if there is a top port to the right of tport then go to step q else go to step r,
q) set tport to that top port left of tport, and go to step l, and
r) end.

13) The row selection method as claimed in step j of Claim 10 as shown in Fig. 16B, where num_row is the number of rows in the redundant multi-stage network, and where row_index is the number of a row in the redundant multi-stage network,

wherein the port selection method comprises the steps:

a) begin
b) if there are an even number of rows then go to step c else go to step f,
c) if the current row index is even then go to step d else go to step e,
d) set return value to num_rows / 2 + row_index/2 – 1 and go to step i,
e) set return value to num_rows / 2 – (row_index+1)/2 and go to step i,
f) if the current row index is even then go to step g else go to step h,
g) set return value to (num_rows – 1)/2 + row_index / 2 –1 and go to step i,
h) set return value to (num_rows – row_index) / 2 – 1 and go to step i, and
i) end

14) The port selection method as claimed in step k of Claim 10 as shown in Fig. 17B,

wherein the port selection method comprises the steps:

a) begin,
b) establish scanning order of disconnect ports,
c) set test port to first port in scanning order,
d) find corresponding port to test port,
e) if the corresponding port is connected then go to step f else go to step o,
f) if there is a port next in the scanning order then go to step d else go to step g,
g) set test port to first port in scanning order,
h) find the connection to the corresponding port to test port,
i) if either router connected to this connection already have a disconnected port then go to step j else go to step p,
j) if there is a port next in the scanning order then go step k else go to step l.
k) Set test port to next port in scanning order and go to step h,
l) if there is a disconnected port then go to step q else go to step m,
m) if there is a port not connected to its proper port then go to r else go to step n,
n) report no ports need rewiring and go to step s,
o) set return value to test port and go to step s,
p) set return value to test port and go to step s,
q) set return value to this port and go to step s,
r) set return value to this port and go to step s,
s) end

15) The port selection method as claimed in step k of Claim 10 as shown in Fig. 17A, where test_port is a variable which indicates a port value,

wherein the port selection method comprises the steps:

a) begin,

b) if a bottom port was disconnected in the previous step then go to step h else go to step c,

c) establish scanning order of bottom ports,

d) set test_port to first port in scanning order,

e) if the test_port is connected to its proper port then go to step f else go to step i,

f) if there is a port next in the scanning order then go to step g else go to step j,

g) set test_port to next port in scanning order and go to step e,

h) set return value to the disconnected port and go to step k,

i) set return value to test_port and go to step k,

j) report no ports need rewiring and go to step k, and

k) end.

16) The port selection method as claimed in step k of Claim 10 as shown in Fig. 17C,

wherein the port selection method comprises the steps:

a) begin,
b) get "port fifo" from previous selection call,
c) if the fifo is empty then f else go to step d,
d) set return value to top of fifo,
e) remove top of fifo go to step k,
f) load "port fifo" with all disconnected ports,
g) if the fifo is empty then go to step h else go to step d,
h) if there is a port not connected to its proper port then go to step i else go to step j,
i) set return value to this port and go to step k,
j) report no ports need rewiring and go to step k, and
k) end.